Berechenbarkeit

LOOP-Berechenbar

LOOP x_i DO P END und Zeichen $;$, $:=$, $+$, $-$ (mod. Substrak.)
P wird mit initialem Wert x_i oft ausgeführt. Alle Variablen x_i, $i \in \mathbb{N}$, sind mit 0 initialisiert. x_0 ist Ausgabe. Parameter $f(x_1, x_2, \ldots)$ werden in Var. $x_1, x_2, \ldots$ geschrieben

= Primitive Rek.

Grundfunktionen :

- $k : \mathbb{N}^l \to \mathbb{N}$ konstante Funktion
- $\Pi_i^l : \mathbb{N}^l \to \mathbb{N}$ Projektion auf i-tes Element $(x_1, \ldots, x_l) \to x_i$
- $s(n) = n + 1$ Nachfolger
- Einsetzen mit $g : \mathbb{N}^m \to \mathbb{N},\ h_i : \mathbb{N}^l \to \mathbb{N}$
  $\Rightarrow \mathbb{N}^l \to \mathbb{N}\ (x_1, \ldots, x_l) \mapsto g(h_1(x_1, \ldots, x_l), \ldots, h_m(x_1, \ldots, x_l))$
- Primitive Rekursion : $f(n, x_1, \ldots, x_k) =$
  $\begin{cases} g(x_1, \ldots, x_k), & n = 0 \\ h(f(n-1, x_1, \ldots, x_n), n, x_1, \ldots, x_k), & sonst \end{cases}$
  $g(x) = 0 \quad h(z, n, x) = \text{add}(z, x) \quad \Rightarrow \quad f(y, x) = y \cdot x$

  $\text{even}(0) = 1 = c_1^0$
  $\text{even}(x+1) = \text{zero}(\text{even}(x)) = \text{zero}(\Pi_1^2(\text{even}(x), x))$

$\subsetneq$ WHILE-, GOTO-Berechenbar

(da Ackermannfunktion oder nirgends definierte Funktion nicht LOOP-berch.)
WHILE $x_i \neq 0$ DO P END

= Turing-Berechenbar

TM M_i existiert für $f(n_i, \ldots, n_k) = m$. M_i hält mit m auf Ausgabe-band, wenn Eingabe das Tupel $(n_1, \ldots, n_k)$ war.

= μ-Berechenbar (μ-Rekursiv)

mit $f : \mathbb{N}^{k+1} \to \mathbb{N}$ $\quad \mu f : \mathbb{N}^k \to \mathbb{N}$
$\mu f(x_1, \ldots, x_k) =$
$\min\{\ n \mid f(n, x_1, \ldots, x_k) = 0 \ \land\ \forall m < n : f(m, x_1, \ldots, x_k) > 0\}$
Für $f(x, y) = 2$ ist μf nirgends def.
Sei f μ-Rekursiv.s Dann exist. p, q als $(k+1)$-stellige prim. rekursive Funktionen mit :
$f(x_1, \ldots, x_k) = p(x_1, \ldots, x_k, \mu q(x_1, \ldots, x_k))$
$\sim$ Satz von Kleene
$\Rightarrow$ Eine einzige While-Schleife kann das gleiche berechnen, wie ein Programm mit mehren Schleifen.

= Arithmetisch Repräsentierbar

Terme $t_1, t_2, \ldots$ bilden Formeln zB : $t_1 = t_2$
Formeln : $\neg F, F \land G, \ldots$. Quantoren $\exists, \forall, \in$ erzeugen gebundene Var.
Belegungen mit zB $\Phi(x) = 3,\ \Phi(y) = 3, \ldots$ führen zu wahren/falschen Aussagen $\Phi(F)$
$f : \mathbb{N}^k \to \mathbb{N}$ ist arithmetisch repräsentierbar, falls F existiert mit :
$F(n_1, \ldots, n_k, m) \ \Leftrightarrow\ f(n_1, \ldots, n_k) = m$
$f(x, y) = x \cdot y$ a.r. mit $F(x, y, z) \Leftrightarrow ((x \cdot y) = z)$
Element : $x \geqslant 1 \ \Rightarrow\ \exists k : k + 1 = x$
Danach Korrektheit beweisen : F wahr $\Leftrightarrow \ldots = f$

Church'sche These

Alle diese letzten Modelle beschreiben das gleiche, wie der intuitive Berechenbarkeitsbegriff.

Wachstum

von Programm P werden alle Var aufsummiert :
$f_P(n) = \max\left\{\sum_{i \geqslant 0} n_i' \mid \sum_{n_i \geqslant 0} n_i \leqslant n\right\}$
Bei LOOP : $\exists k : \forall n : f_P(n) < a(k, n)$

Entscheidbarkeit

Menge ist **Entscheidbar**, wenn für Menge A charakteristische Funktion χ_A berechenbar ist. $\chi_A(w) = \begin{cases} 1, & w \in A \\ 0, & w \notin A \end{cases}$

A entscheidbar $\Leftrightarrow A, \bar{A}$ semi-entscheidbar
Menge ist **Semi-Entscheidbar**, wenn χ_A' wahr für $w \in A$ zurück gibt (also hält). $\chi_A(w) = \begin{cases} 1, & w \in A \\ \text{undef.}, & w \notin A \end{cases}$

Semi-Entscheidbar ist äquivalent zu :

- **rekursiv Aufzählbar** : $\exists f : \mathbb{N} \to \Sigma^* \ :\ A = \{f(n) | n \in \mathbb{N}\}$
- A ist Typ 0
- $\exists$ Turing Maschine M : $A = T(M)$
- χ' ist berechenbar
- A ist Definitions- oder Zielbereich von berechenbarer Funktion.

Halteproblem

spezielles Halteproblem $K = \{w \in \{0,1\}^* \mid M_w$ hält auf Eingabe $w\}$
Halteproblem $H = \{w\#x \mid M_w$ hält auf Eingabe $x\}$ Halteproblem
$H_0 = \{w \mid M_w$ hält auf Eingabe $\epsilon\}$

Satz von Rice

Nicht-triviale Aussagen über die Spracheigenschaften von TM sind unentscheidbar.
$S \subseteq R$ turing-berechenbare Funk. mit $\varnothing \neq S \neq R$
$C(S) = \{w \mid M_w$ berechnet eine Funktion aus $S\}$ ist unentscheidbare Menge.
Nur für Sprachen verwenden, deren Elemente kodierte TMs sind!
Verwendung im Beweis :
- Zeigen : Sprache ist semantisch (z.B : Hängt nur von $T(M)$ ab)
- Zeigen : Sprache ist nicht trivial. (Beispiele von Eingaben, für die Sprache jeweils wahr/falsch)

Komplexität

Alle deterministischen Klassen sind gegen Komplement abgeschlossen.
zB : $A \in \text{DTIME}(\mathcal{O}(\log(n)))$

Zeitklassen

DTIME ist gegen Komplement abgeschlossen
NTIME nicht.

P - *Polynomialzeit*

durch LOOP-Programme entscheidbar

NP - *Nichtdeterministische Polynomialzeit*

NP-hart $\forall L \in \text{NP}$: $L \leqslant_p A$
NP-vollständig NP-hart und Sprache $A \in \text{NP}$
$A \leqslant_p B \ \land\ B \in (\text{N})\text{P} \Rightarrow A \in (\text{N})\text{P}$
Beweis $A \in \text{NP}$ oft mit guess & check

EXPTIME - *Exponentialzeit*

$2^{p(n)}$ mit Polynom p

Platzklassen

SPACE und NSPACE sind gegen Komplement abgeschlossen,
wenn $f \in \Omega(\log(n)) \quad \Rightarrow$ NSPACE(f) = coNSPACE(f)
(NSPACE : Immerman und Szelepcsényi)

Wird x viel Platz nicht verlassen, endet die Maschine nach spätestens $|x| \cdot |Z| \cdot |\Gamma|^{(|x|+1)}$ in einer Schleife

L = LOGSPACE - *logarithmischer Platz*

NL - *nichtdeterministischer log. Platz*

PSPACE - *polynomieller Platz*

$= \bigcup_{k \geqslant 1} \text{DSPACE}(n^k) = \bigcup_{k \geqslant 1} \text{NSPACE}(n^k)$

Zeit- / Platzrelationen

$\text{DTIME}(f) \subseteq \text{DTIME}_{2-\text{Band}}(f \log f)$
$\sim$ Satz von Hennie und Stearns (wenn $\epsilon > 0$ mit $\forall n : f(n) \geqslant (1+\epsilon)n$ existiert)

$\text{DNSPACE}(f) = \text{DNSPACE}_{1-\text{Band}}(f)$

$\text{DTIME}(f) = \text{DTIME}(\mathcal{O}(f))$ falls $\exists \epsilon > 0 : \forall n :\ f(n) \geqslant (1+\epsilon)n$
$\text{NTIME}(f) = \text{NTIME}(\mathcal{O}(f))$ immer
$\sim$ Zeit-/Platz-kompressionssatz

$\text{DSPACE}(4^{\mathcal{O}(n)}) = \bigcup_{c \in \mathbb{N}} \text{DSPACE}(4^{c \cdot n})$

$\text{DTIME}(f) \subseteq \text{NTIME}(f) \subseteq \text{DSPACE}(f) \quad \forall f : \mathbb{N} \to \mathbb{N}, \forall n : f(n) \geqslant n$
$\text{DSPACE}(f) \subseteq \text{NSPACE}(f) \subseteq \text{DTIME}(2^{\mathcal{O}(f)}) \quad \forall f : \mathbb{N} \to \mathbb{N},$
$\forall n : f(n) \geqslant \log(n)$ - exponentieller Blowup
$\Rightarrow \text{DSPACE}(f) \subseteq \text{DTIME}(2^{\mathcal{O}(f)})$

$\text{NSPACE}(s) \subseteq \text{DSPACE}(s^2)$ mit $s \in \Omega(\log n)$
$\sim$ **Satz von Savitch**

Zeit- / Platzkonstruierbar

Deterministische TM existiert, die bei unär kodierter Eingabe a^n der Länge n, $f(n)$ viel Platz/Zeit braucht.

___-Hierarchiesatz

Platz : Sei
$s_1, s_2 : \mathbb{N} \to \mathbb{N}$, $s_1 \notin \Omega(s_2)$, $s_2 \in \Omega(\log n)$, s_2 platzkonstruierbar
$\text{DSPACE}(s_2) \backslash \text{DSPACE}(s_1) \neq \varnothing$
Beweis für $s_1 \notin \Omega(s_2)$: $\quad \forall c > 0 : \exists a \in \mathbb{N} : s_1(a) < c \cdot s_2(b)$ Aufstellen und a suchen, für das Gleichung stimmt.
$\Rightarrow \quad \text{DSPACE}(\log) \subsetneq \text{DSPACE}(\log^2)$

Zeit : Sei
$t_1, t_2 : \mathbb{N} \to \mathbb{N}$, $t_1 \log(t_1) \notin \Omega(t_2)$, $t_2 \in \Omega(n \log(n))$, t_2 zeitkonstruierbar
$\text{DTIME}(t_2) \backslash \text{DTIME}(t_1) \neq \varnothing$
$\Rightarrow \quad \text{DTIME}(\mathcal{O}(n)) \subsetneq \text{DTIME}(\mathcal{O}(n^2))$

Sei r total und berechenbar. $\forall n : r(n) \geqslant n$
$\Rightarrow \exists$ totale Funktion $s : \mathbb{N} \to \mathbb{N} \quad s(n) \geqslant n + 1$ mit
$\text{DTIME}(s) = \text{DTIME}(r \circ s)$
$\sim$ Satz von Borodim
(s ist nicht zeitkonstruierbar)

Probleme

Zeit, Platz

PCP - *Post-Korrespondenz-Problem*

χ_{PCP} ist berechenbar $\Rightarrow$ PCP ist rek. aufzählbar. $\Leftrightarrow$ semi-entscheidbar.
PCP ist aber unentscheidbar (für $\Sigma \geqslant 2$) H $\leqslant$ MPCP $\leqslant$ PCP

SAT - *Satisfiablity*

NP-vollständig
SAT = {F | F ist erfüllbar}
Algos aktuell bei $2^{c \cdot n}$ (also $\in$ E)
Allgemein äquivalent zu
3KNF-SAT, beide NP-vollständig
2KNF-SAT $\in$ P & NL-vollständig

CLIQUE

NP-vollständig
Graph $G = (V, E), \quad k \in \mathbb{N}$
$V' \subseteq V$ ist Clique, falls $\forall u, v \in V' : \ u \neq v \ \Rightarrow \ (u, v) \in E$
CLIQUE $\in$ NP durch guess & check.
NP-Vollständigkeit durch 3KNF-SAT$\leqslant$CLIQUE mittels Graph mit $E = \{\{(r, s), (p, q)\} | r \neq p \wedge z_{rs} \not\equiv \neg z_{pq}\}$ (Alle Literale, die sich nicht gegenseitig ausschließen)

FÄRBBARKEIT

NP-vollständig
$\varphi : V \to \{1, ..., k\}$ für Graph $G = (V, E), \quad k \in \mathbb{N}$,
Knotenfärbung mit k Farben : $\forall (u, v) \in E : \ \varphi(u) \neq \varphi(v)$
FÄRBBARKEIT $\in$ NP durch guess & check.
NP-Vollständigkeit durch 3KNF$\leqslant$3-Färbbarkeit

GAP - *Grapherreichbarkeit*

$\in$P, NL-vollständig
Auf Graph $G = (V, E), \quad k \in \mathbb{N}$ und 2 Knoten : $s, t \in V$
Kann man über Kanten $\in E$ von s zu t gelangen?
DSPACE($\log^2 n$)
GAP $\in$ NL, da :
WHILE $v \neq t$ DO {
Wähle nicht-det. $w \in V, \quad$ aus $(v, w) \in E$
$v = w$ }
RETURN 1

CVP - *Circuit Value Problem*

P-vollständig Bei Schaltkreisen können (Teil)formeln wiederverwendet werden.

QBF - *Quantifizierbare Boolsche Formeln*

PSPACE-vollständig

TSP - *Traveling Salesman Problem*

$\in$NP

VC - *Vertex Cover*

Allgemeines

TM $M = (Z, \Sigma, \Gamma, \delta, z, \square, E)$ - Z : Zustandsmenge, Γ : Bandalphabet, Übertragungsfunktion $\delta(z_i, a) = (z_j, a', \mathrm{L})$, z : Startzustand
$a, a' \in \Gamma$, statt L auch L,R,N
Sprache T(M)
TM ist äquivalent zu Mehrband-TMs und nicht det. TM

Grammatik : $G = (V, \Sigma, P, S)$ mit $P \subseteq (V \cup \Sigma)^+ \times (V \cup \Sigma)^*$

Disjunktion : $\vee$, Konjunktion $\wedge$, DNF : $\bigvee_i \bigwedge_j (\neg) x_{ij}$
Bestimmte Verknüpfung der Unterterme.
$\dot{-}$ Modifizierte Differenz : $\max\{0, a - b\} = \mathrm{md}(a, b)$
Belegung $\mathscr{A}$ passt zu Formel F, wenn jede vorkommende atomare Variable einen Wert zugewiesen bekommt.
Belegung $\mathscr{A}$ ist Modell, wenn passend und $\mathscr{A}(F) = 1$.
Formel F ist gültig, wenn für alle $\mathscr{A}$, die zu F passen, $\mathscr{A}(F) = 1$ gilt (Tautologie). "Ungültig" existiert nicht
$\binom{n}{k} = \dfrac{n!}{(n-k)! \cdot k}$
Nirgends definierte Funktion Σ (berechenbar)

Ackermann-Funktion $\mathbf{A}(n) = \mathbf{a}(n, m)$

$$\begin{aligned} \mathrm{a}(0, y) &= y + 1 \\ \mathrm{a}(x, 0) &= \mathrm{a}(x - 1, 1) \qquad x, y > 0 \\ \mathrm{a}(x, y) &= \mathrm{a}(x - 1, \mathrm{a}(x, y - 1)) \end{aligned}$$

$$\begin{aligned} y &< \mathrm{a}(x, y) \\ \mathrm{a}(x, y) &< \mathrm{a}(x, y + 1) \\ \mathrm{a}(x, y + 1) &\leqslant \mathrm{a}(x + 1, y) \\ \mathrm{a}(x, y) &< \mathrm{a}(x + 1, 1) \end{aligned}$$

Bijektion $\mathbb{N}^2 \to \mathbb{N}$

Kodieren von Tupeln :
$\mathrm{c}(x, y) = \binom{x+y+1}{2} + x = \mathrm{add}(f(s(\mathrm{add}(x, y))), x)$

Dove-Tailing

— $\Sigma^* = \{w_1, w_2, ...\}$ Längenlexikographisch Anordnen
— FOR $i = 0, 1, 2, ...$ DO
Simuliere i Schritte von M_w auf Eingabe $e(i)$
...Kriterium...

Translationstechnik

Padding einer Sprache mit $\$ \notin \Sigma$
$\mathrm{Pad}_f(L) = \{w\$^{f(|w|) - |w|} \mid w \in L\}$
Zeit : $\mathrm{Pad}_f(L) \in \mathrm{DNTIME}(\mathcal{O}(g)) \Leftrightarrow \mathrm{L} \in \mathrm{DNTIME}(\mathcal{O}(g \circ f))$
mit f, g zeitkonstruierbar, $g(n), f(n) \geqslant n$
Platz : $\mathrm{Pad}_f(L) \in \mathrm{DNSPACE}(\mathcal{O}(g)) \Leftrightarrow \mathrm{L} \in \mathrm{DNSPACE}(\mathcal{O}(g \circ f))$
mit $g \in \Omega(\log)$, $\forall n : f(n) \geqslant n$ berechenbar

$\Rightarrow \mathrm{DSPACE}(n) \neq \mathrm{P}$

Aufzählbar / Abzählbar

rekursiv Aufzählbar	**Abzählbar**
totale Funktion $f : \mathbb{N} \to \Sigma^*$	
f berechenbar	

Mit $A \subseteq B$ folgt NICHT :
B rekursiv aufzählbar $\not\Rightarrow$ A rek. aufzählbar. Nur A abzählbar

Reduktion

A ist auf B reduzierbar $A \leqslant B$, wenn totale & berechenbare Funktion $f : \Sigma^* \to \Gamma^*$ existiert mit :
$x \in A \Leftrightarrow f(x) \in B$

$\leqslant$ unbeschränkt $\quad \leqslant_p$ polynomialzeit $\quad \leqslant_{log}$ Logspace (f ist logspace-berechenbar)

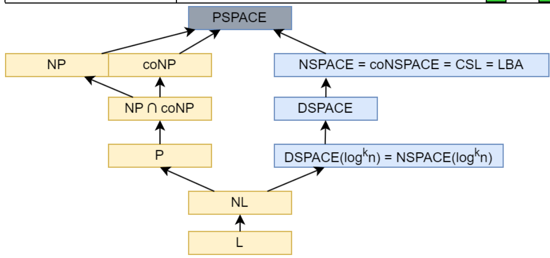
$A \leqslant B$ und B (semi-)entscheidbar $\Rightarrow$ A (semi-)entscheidbar

Landau-Symbole

$f \in \mathcal{O}$: f wächst nicht wesentlich schneller als ...
$f \in \Omega$: f wächst nicht wesentlich langsamer als ...
Beweis $f(n) \in \mathcal{O}(b(n)) : \exists c \exists n_0 \ \forall (n \geqslant n_0) : \ b(n) \leqslant c \cdot f(n)$

Relationen

	$L_1 = L_2$	$L_1 \subseteq L_2$	$L_1 \cap L_2 = \emptyset$	$\lvert L_1 \cap L_2 \rvert < \infty$	$L(G) = L_1 \cap L_2$ G ist Typ-2	Schnitt $\cap$	Verein. $\cup$	Kompl.	Produkt	Stern
Typ-3 Reg			🟩			🟩	🟩	🟩	🟩	🟩
DCFL Det. Kntx-frei	🟩	🟥	🟥	🟥	🟥	🟥	🟥	🟩	🟥	🟥
Typ-2 CFL Kntx-frei	🟥	🟥	🟥	🟥	🟥	🟥	🟩	🟥	🟩	🟩
Typ-1 CSL						🟩	🟩	🟩	🟩	🟩
Typ-0						🟩	🟩	🟥	🟩	🟩



L $\subseteq$ NL $\subseteq$ P $\subseteq$ PSPACE $\subseteq$ EXPTIME
$\Rightarrow \mathrm{DSPACE}(\log n) \subseteq \mathrm{NSPACE}(\log n) \subseteq \mathrm{DTIME}(2^{\mathcal{O}(\log n)}) = \mathrm{P}$

$G(a, b, i, \cdot)$-Prädikat

$G(a, b, i, y) \Leftrightarrow y = a \mod (1 + (i + 1) \cdot b)$
$\Rightarrow \ y \leqslant (i + 1)b \quad (1 + (i + 1)b \ \% \ (a - y)) = 0$
a, b zwei Werte, die endliche Folge $(n_0, ..., n_k)$ kodieren.
i Index, y Wert. Es gilt für alle $i \leqslant k$:
$n_i = y \ \Leftrightarrow \ G(a, b, i, y)$
$\forall k \forall (n_0, ..., n_k) \in \mathbb{N}^{k+1} \ \exists a, b \in \mathbb{N} \ \forall i \in \{0, ..., k\} : \ G(a, b, i, n_i)$